

09-25-00

PTO/SB/05 (4/95)

Approved for use through 09/30/2000 CMB 0651-0032

Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
A collection of information unless it displays a valid OMB control number.

Please type a plus sign (+) inside this box

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

UTILITY
PATENT APPLICATION
TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No. 042390.P8628X

First Inventor or Application Identifier Carl M. Ellison

Title Managing a Secure Environment Using a Chipset in Isolated Execution Mode

Express Mail Label No. EL46631468US

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents

ADDRESS TO:

Assistant Commissioner for Patents
Box Patent Application
Washington, DC 20231

1. ☒ Free Transmittal Form
(Submit an original, and a duplicate for fee processing)
2. ☒ Specification [Total Pages 28]
(preferred arrangement set forth below)
- Descriptive title of the Invention
 - Cross References to Related Applications
 - Statement Regarding Fed sponsored R & D
 - Reference to Microfiche Appendix
 - Background of the Invention
 - Brief Summary of the Drawings (if filed)
 - Detailed Description
 - Claim(s)
 - Abstract of the Disclosure
3. ☒ Drawing(s) (35 U.S.C. 113) [Total Sheets 8]
4. Oath or Declaration [Total Pages 12]
- a. ☒ Newly executed (original copy)
- b. ☐ Copy from a prior application (37 C.F.R. § 1.63(d))
(for continuation/divisional with Box 16 completed)
- i. ☐ **DELETION OF INVENTOR(S)**
Signed statement attached deleting inventor(s) named in the prior application,
see 37 CFR §§ 1.63(d)(2) and 1.33(b).

NOTE FOR ITEMS 1 & 13 IN ORDER TO BE ENTITLED TO PAY SMALL
ENTITY FEES, A SMALL ENTITY STATEMENT IS REQUIRED (37 C.F.R.
§ 1.27), EXCEPT IF ONE FILED IN A PRIOR APPLICATION IS RELIED
UPON (37 C.F.R. § 1.28).

5. ☐ Microfiche Computer Program (Appendix)
6. Nucleotide and/or Amino Acid Sequence Submission
(if applicable, all necessary)
- a. ☐ Computer Readable Copy
- b. ☐ Paper Copy (identical to computer copy)
- c. ☐ Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

7. ☒ Assignment Papers (cover sheet & document(s))
8. ☐ 37 C.F.R. § 3.73(b) Statement (when there is an assignee) ☐ Power of Attorney
9. ☐ English Translation Document (if applicable)
10. ☐ Information Disclosure Statement (IDS)/PTO - 1449 ☐ Copies of IDS Citations
11. ☐ Preliminary Amendment
12. ☒ Return Receipt Postcard (MPEP 503)
(Should be specifically itemized)
13. ☐ *Small Entity Statement(s) ☐ Statement filed in prior application, Status still proper and desired
14. ☐ Certified Copy of Priority Document(s)
(if foreign priority is claimed)
15. Other:

16. If a CONTINUING APPLICATION, check appropriate box, and supply the requisite information below and in a preliminary amendment:
☐ Continuation ☐ Divisional ☒ Continuation-in-part (CIP) of prior application No: 09/540.613

Prior application information: Examiner Not Assigned Group/Art Unit:

For CONTINUATION or DIVISIONAL APPS only: The entire disclosure of the prior application, from which an oath or declaration is supplied under Box 4b, is considered a part of the disclosure of the accompanying continuation or divisional application and is hereby incorporated by reference. The incorporation can only be relied upon when a portion has been inadvertently omitted from the submitted application parts.

17. CORRESPONDENCE ADDRESS

☐ Customer Number of Bar Code Label

(Insert Customer No. or Attach bar code label here)

or ☒ Correspondence address below

Name

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Address

12400 Wilshire Boulevard, Seventh Floor

City

Los Angeles

State

California

Zip Code

90025

Country

U.S.A.

Telephone

(714) 557-3800

Fax

(714) 557-3347

Name (Print/Type)

Thinh V. Nguyen, Reg. No. 42,034

Date

09/22/00

Signature

Thinh V. Nguyen

Burden Hour Statement: This form is obligated to be completed by the inventor(s) or applicant(s). Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.

Please type a plus sign (+) inside this box +

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number

PTO/SB/17 (12/99)

Approved for use through 09/30/2000 OMB 0651-0032
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

FEE TRANSMITTAL for FY 2000

Patent fees are subject to annual revision.
Small entity payments must be supported by a small entity statement,
otherwise large entity fees must be paid. See Forms PTO/SB/09-12.
See 37 C.F.R. §§ 1.27 and 1.28.

TOTAL AMOUNT OF PAYMENT (\$) 1,528.00

Complete if Known

Application Number
Filing Date September 22, 2000
First Named Inventor Carl M. Ellison
Examiner Name Not Assigned
Group/Art Unit
Attorney Docket No. 042390.P8628X

METHOD OF PAYMENT (check one)

1. ☒ The Commissioner is hereby authorized to charge indicated fees to
☒ The Commissioner is hereby authorized to credit any over payments to:

Deposit Account Number 02-2666
Deposit Account Name Blakely, Sokoloff, Taylor & Zafman LLP

☒ Charge Any Additional Fees Required Under 37 CFR §§ 1.16, 1.17, 1.18 and 1.20.

2. ☒ Payment Enclosed:
☒ Check ☐ Money Order ☐ Other

FEE CALCULATION

BASIC FILING FEE

Large Entity	Small Entity	Fee Code	Fee (\$)	Fee Description	Fee Paid
101	690	201	345	Utility filing fee	\$690.00
106	310	206	155	Design filing fee	
107	480	207	240	Plant filing fee	
108	690	208	345	Reissue filing fee	
114	150	214	75	Provisional filing fee	
SUBTOTAL (1)					\$ 690.00

EXTRA CLAIM FEES

Large Entity	Small Entity	Fee Code	Fee (\$)	Fee Description	Fee Paid
103	18	203	9	Claims in excess of 20	
102	78	202	39	Independent claims in excess of 3	
104	260	204	130	Multiple Dependent claim, if not paid	
109	78	209	39	**Reissue independent claims over original patent	
110	18	210	9	**Reissue claims in excess of 20 over original patent	
SUBTOTAL (2)					\$ 798.00

*or number previously paid, if greater. For Reissues, see below

Large Entity	Small Entity	Fee Code	Fee (\$)	Fee Description	Fee Paid
103	18	203	9	Claims in excess of 20	
102	78	202	39	Independent claims in excess of 3	
104	260	204	130	Multiple Dependent claim, if not paid	
109	78	209	39	**Reissue independent claims over original patent	
110	18	210	9	**Reissue claims in excess of 20 over original patent	
SUBTOTAL (2)					\$ 798.00

3. ADDITIONAL FEE

Large Entity	Small Entity	Fee Code	Fee (\$)	Fee Description	Fee Paid
105	130	205	65	Surcharge - late filing fee or oath	
127	50	227	25	Surcharge - late provisional filing fee or cover sheet	
139	130	139	130	Non-English specification	
147	2,520	147	2,520	For filing a request for reexamination	
112	920*	112	920*	*Requesting publication of SIR prior to Examiner action	
113	1,840*	113	1,840*	*Requesting publication of SIR after Examiner action	
115	110	215	55	Extension for response within first month	
116	380	216	190	Extension for response within second month	
117	870	217	435	Extension for response within third month	
118	1,210	218	680	Extension for response within fourth month	
128	1,850	228	925	Extension for response within fifth month	
119	300	219	150	Notice of Appeal	
120	300	220	150	Filing a brief in support of an appeal	
121	260	221	130	Request for oral hearing	
138	1,510	138	1510	Petition to institute a public use proceeding	
140	110	240	55	Petition to revive - unavoidable	
141	1,210	241	605	Petition to revive - unintentional	
142	1,210	242	605	Utility issue fee (or reissue)	
143	430	243	215	Design issue fee	
144	580	244	290	Plant issue fee	
122	130	122	130	Petitions to the Commissioner	
123	50	123	50	Petitions related to provisional applications	
126	240	126	240	Submission of Information Disclosure Stmt	
581	40	581	40	Recording each patent assignment per property (times number of properties)	40.00
146	790	246	395	Filing a submission after final rejection (37 CFR 1.129(a))	
149	790	249	395	For each additional invention to be examined (37 CFR 1.129(b))	
Other fee (specify)					
Other fee (specify)					
* Reduced by Basic Filing Fee Paid					
SUBTOTAL (3)					\$ 40.00

SUBMITTED BY

Typed or Printed Name **Thinh V. Nguyen**
Signature *Thinh V. Nguyen*

Date

09/22/00

Deposit Account User ID

02-2666

Complete (if applicable)

Business Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete the form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.

UNITED STATES PATENT APPLICATION

FOR

**MANAGING A SECURE ENVIRONMENT USING A CHIPSET IN ISOLATED
EXECUTION MODE**

INVENTORS:

CARL M. ELLISON
ROGER A. GOLLIVER
HOWARD C. HERBERT
DERRICK C. LIN
FRANCIS X. MCKEEN
GIL NEIGER
KEN RENERIS
JAMES A. SUTTON
SHREEKANT S. THAKKAR
MILLAND MITTAL

PREPARED BY:

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP
12400 Wilshire Blvd., 7th Floor
Los Angeles, CA 90025-1026
(714) 557-3800

CROSS-REFERENCES TO RELATED APPLICATIONS

This is a continuation-in-part of U.S. Patent Application No. 09/540,613 filed March 31, 2000.

BACKGROUND5 1. FIELD OF THE INVENTION

This invention relates to microprocessors. In particular, the invention relates to processor security.

2. DESCRIPTION OF RELATED ART

Advances in microprocessor and communication technologies have opened up
10 many opportunities for applications that go beyond the traditional ways of doing
business. Electronic commerce (E-commerce) and business-to-business (B2B)
transactions are now becoming popular, reaching the global markets at a fast rate.
Unfortunately, while modern microprocessor systems provide users convenient and
efficient methods of doing business, communicating and transacting, they are also
15 vulnerable to unscrupulous attacks. Examples of these attacks include theft of data,
virus, intrusion, security breach, and tampering, to name a few. Computer security,
therefore, is becoming more and more important to protect the integrity of the computer
systems and increase the trust of users.

Threats caused by unscrupulous attacks may be in a number of forms. Attacks
20 may be remote without requiring physical accesses. An invasive remote-launched
attack by hackers may disrupt the normal operation of a system connected to thousands
or even millions of users. A virus program may corrupt code and/or data of a single-
user platform.

Existing techniques to protect against attacks have a number of drawbacks.
25 Anti-virus programs can only scan and detect known viruses. Most anti-virus programs
use a weak policy in which a file or program is assumed good until proved bad. For
many security applications, this weak policy may not be appropriate. In addition, most
anti-virus programs are used locally where they are resident in the platform. This may
not be suitable in a group work environment. Security co-processors or smart cards
30 using cryptographic or other security techniques have limitations in speed performance,

memory capacity, and flexibility. Redesigning operating systems creates software compatibility issues and causes tremendous investment in development efforts.

BRIEF DESCRIPTION OF THE DRAWINGS

The features and advantages of the present invention will become apparent from the following detailed description of the present invention in which:

5 Figure 1A is a diagram illustrating a logical operating architecture according to one embodiment of the invention.

Figure 1B is a diagram illustrating accessibility of various elements in the operating system and the processor according to one embodiment of the invention.

Figure 1C is a diagram illustrating a computer system in which one embodiment of the invention can be practiced.

10 Figure 2 is a diagram illustrating a chipset environment according to one embodiment of the invention.

Figure 3 is a diagram illustrating a chipset circuit shown in Figure 2 according to one embodiment of the invention.

15 Figure 4 is a flowchart illustrating a process to manage a chipset according to one embodiment of the invention.

Figure 5 is a flowchart illustrating a process to enroll a thread according to one embodiment of the invention.

Figure 6 is a flowchart illustrating a process to withdraw a thread according to one embodiment of the invention.

DETAILED DESCRIPTION

In the following description, for purposes of explanation, numerous details are set forth in order to provide a thorough understanding of the present invention. However, it will be apparent to one skilled in the art that these specific details are not required in order to practice the present invention. In other instances, well-known electrical structures and circuits are shown in block diagram form in order not to obscure the present invention.

ARCHITECTURE OVERVIEW

One principle for providing security in a computer system or platform is the concept of an isolated execution architecture. The isolated execution architecture includes logical and physical definitions of hardware and software components that interact directly or indirectly with an operating system of the computer system or platform. An operating system and the processor may have several levels of hierarchy, referred to as rings, corresponding to various operational modes. A ring is a logical division of hardware and software components that are designed to perform dedicated tasks within the operating system. The division is typically based on the degree or level of privilege, namely, the ability to make changes to the platform. For example, a ring-0 is the innermost ring, being at the highest level of the hierarchy. Ring-0 encompasses the most critical, privileged components. In addition, modules in Ring-0 can also access to lesser privileged data, but not vice versa. Ring-3 is the outermost ring, being at the lowest level of the hierarchy. Ring-3 typically encompasses users or applications level and executes the least trusted code. It is noted that the level of the ring hierarchy is independent to the level of the security protection of that ring.

Figure 1A is a diagram illustrating a logical operating architecture 50 according to one embodiment of the invention. The logical operating architecture 50 is an abstraction of the components of an operating system and the processor. The logical operating architecture 50 includes ring-0 10, ring-1 20, ring-2 30, ring-3 40, and a processor nub loader 52. The processor nub loader 52 is an instance of a processor executive (PE) handler. The PE handler is used to handle and/or manage a processor executive (PE) as will be discussed later. The logical operating architecture 50 has two modes of operation: normal execution mode and isolated execution mode. Each ring in the logical operating architecture 50 can operate in both modes. The processor nub loader 52 operates only in the isolated execution mode.

Ring-0 10 includes two portions: a normal execution Ring-0 11 and an isolated execution Ring-0 15. The normal execution Ring-0 11 includes software modules that are critical for the operating system, usually referred to as kernel. These software modules include primary operating system (e.g., kernel) 12, software drivers 13, and hardware drivers 14. The isolated execution Ring-0 15 includes an operating system (OS) nub 16 and a processor nub 18. The OS nub 16 and the processor nub 18 are instances of an OS executive (OSE) and processor executive (PE), respectively. The OSE and the PE are part of executive entities that operate in a secure environment associated with the isolated area 70 and the isolated execution mode. The processor nub loader 52 is a protected bootstrap loader code held within a chipset in the system and is responsible for loading the processor nub 18 from the processor or chipset into an isolated area as will be explained later.

Similarly, ring-1 20, ring-2 30, and ring-3 40 include normal execution ring-1 21, ring-2 31, ring-3 41, and isolated execution ring-1 25, ring-2 35, and ring-3 45, respectively. In particular, normal execution ring-3 includes N applications 42₁ to 42_N and isolated execution ring-3 includes K applets 46₁ to 46_K.

One concept of the isolated execution architecture is the creation of an isolated region in the system memory, referred to as an isolated area, which is protected by both the processor and chipset in the computer system. Portions of the isolated region may also be in cache memory. Access to this isolated region is permitted only from a front side bus (FSB) of the processor, using special bus (e.g., memory read and write) cycles, referred to as isolated read and write cycles. The special bus cycles are also used for snooping. The isolated read and write cycles are issued by the processor executing in an isolated execution mode when accessing the isolated area. The isolated execution mode is initialized using a privileged instruction in the processor, combined with the processor nub loader 52. The processor nub loader 52 verifies and loads a ring-0 nub software module (e.g., processor nub 18) into the isolated area. The processor nub 18 provides hardware-related services for the isolated execution.

One task of the processor nub loader 52 and processor nub 18 is to verify and load the ring-0 OS nub 16 into the isolated area, and to generate the root of a key hierarchy unique to a combination of the platform, the processor nub 18, and the operating system nub 16. The operating system nub 16 provides links to services in the primary OS 12 (e.g., the unprotected operating system), provides page management within the isolated area, and has the responsibility for loading ring-3 application

modules 45, including applets 46₁ to 46_K, into protected pages allocated in the isolated area. The operating system nub 16 may also load ring-0 supporting modules.

The operating system nub 16 may choose to support paging of data between the isolated area and ordinary (e.g., non-isolated) memory. If so, then the operating system nub 16 is also responsible for encrypting and hashing the isolated area pages before evicting the page to the ordinary memory, and for checking the page contents upon restoration of the page. The isolated mode applets 46₁ to 46_K and their data are tamper-resistant and monitor-resistant from all software attacks from other applets, as well as from non-isolated-space applications (e.g., 42₁ to 42_N), drivers and even the primary operating system 12. The software that can interfere with or monitor the applet's execution is the processor nub loader 52, processor nub 18 or the operating system nub 16.

Figure 1B is a diagram illustrating accessibility of various elements in the operating system 10 and the processor according to one embodiment of the invention. For illustration purposes, only elements of ring-0 10 and ring-3 40 are shown. The various elements in the logical operating architecture 50 access an accessible physical memory 60 according to their ring hierarchy and the execution mode.

The accessible physical memory 60 includes an isolated area 70 and a non-isolated area 80. The isolated area 70 includes applet pages 72 and nub pages 74. The non-isolated area 80 includes application pages 82 and operating system pages 84. The isolated area 70 is accessible only to elements of the operating system and processor operating in isolated execution mode. The non-isolated area 80 is accessible to all elements of the ring-0 operating system and to the processor.

The normal execution ring-0 11 including the primary OS 12, the software drivers 13, and the hardware drivers 14, can access both the OS pages 84 and the application pages 82. The normal execution ring-3, including applications 42₁ to 42_N, can access only to the application pages 82. Generally applications can only access to their own pages, however, the OS typically provides services for sharing memory in controlled methods. Both the normal execution ring-0 11 and ring-3 41, however, cannot access the isolated area 70.

The isolated execution ring-0 15, including the OS nub 16 and the processor nub 18, can access to both of the isolated area 70, including the applet pages 72 and the nub pages 74, and the non-isolated area 80, including the application pages 82 and the OS pages 84. The isolated execution ring-3 45, including applets 46₁ to 46_K, can access

only applet pages 72. The applets 46₁ to 46_K reside in the isolated area 70. In general, applets can only access their own pages; however, the OS nub 16 can also provides services for the applet to share memory (e.g., share memory with other applets or with non-isolated area applications).

5 Figure 1C is a diagram illustrating a computer system 100 in which one embodiment of the invention can be practiced. The computer system 100 includes a processor 110, a host bus 120, a memory controller hub (MCH) 130, a system memory 140, an input/output controller hub (ICH) 150, a non-volatile memory, or system flash, 160, a mass storage device 170, input/output devices 175, a token bus 180, a
10 motherboard (MB) token 182, a reader 184, and a token 186. The MCH 130 may be integrated into a chipset that integrates multiple functionalities such as the isolated execution mode, host-to-peripheral bus interface, memory control. Similarly, the ICH 150 may also be integrated into a chipset together or separate from the MCH 130 to perform I/O functions. For clarity, not all the peripheral buses are shown. It is
15 contemplated that the system 100 may also include peripheral buses such as Peripheral Component Interconnect (PCI), accelerated graphics port (AGP), Industry Standard Architecture (ISA) bus, and Universal Serial Bus (USB), etc. The "token bus" may be part of the USB bus, e.g., it may be hosted on the USB bus.

20 The processor 110 represents a central processing unit of any type of architecture, such as complex instruction set computers (CISC), reduced instruction set computers (RISC), very long instruction word (VLIW), or hybrid architecture. In one embodiment, the processor 110 is compatible with an Intel Architecture (IA) processor, such as the PentiumTM series, the IA-32TM and the IA-64TM. The processor 110 includes a normal execution mode 112 and an isolated execution circuit 115. The
25 normal execution mode 112 is the mode in which the processor 110 operates in a non-secure environment, or a normal environment without the security features provided by the isolated execution mode. The isolated execution circuit 115 provides a mechanism to allow the processor 110 to operate in an isolated execution mode. The isolated execution circuit 115 provides hardware and software support for the isolated execution
30 mode. This support includes configuration for isolated execution, definition of an isolated area, definition (e.g., decoding and execution) of isolated instructions, generation of isolated access bus cycles, and access checking.

In one embodiment, the computer system 100 can be a single processor system, such as a desktop computer, which has only one main central processing unit, e.g.

processor 110. In other embodiments, the computer system 100 can include multiple processors, e.g. processors 110, 110a, 110b, etc., as shown in Figure 1C. Thus, the computer system 100 can be a multi-processor computer system having any number of processors. For example, the multi-processor computer system 100 can operate as part
5 of a server or workstation environment. The basic description and operation of processor 110 will be discussed in detail below. It will be appreciated by those skilled in the art that the basic description and operation of processor 110 applies to the other processors 110a and 110b, shown in Figure 1C, as well as any number of other processors that may be utilized in the multi-processor computer system 100 according
10 to one embodiment of the present invention.

The processor 110 may also have multiple logical processors. A logical processor, sometimes referred to as a thread, is a functional unit within a physical processor having an architectural state and physical resources allocated according to some partitioning policy. Within the context of the present invention, the terms
15 "thread" and "logical processor" are used to mean the same thing. A multi-threaded processor is a processor having multiple threads or multiple logical processors. A multi-processor system (e.g., the system comprising the processors 110, 110a, and 110b) may have multiple multi-threaded processors.

The host bus 120 provides interface signals to allow the processor 110 or
20 processors 110, 110a, and 110b to communicate with other processors or devices, e.g., the MCH 130. In addition to normal mode, the host bus 120 provides an isolated access bus mode with corresponding interface signals for memory read and write cycles. The isolated access bus mode is asserted on memory accesses initiated while the processor 110 is in the isolated execution mode and it is accessing memory within
25 the isolated area. The isolated access bus mode is also asserted on instruction pre-fetch and cache write-back cycles if the address is within the isolated area address range. The isolated access bus mode is configured within the processor 110. The processor 110 responds to a snoop cycle to a cached address when the isolated access bus mode on the FSB matches the mode of the cached address.

30 The MCH 130 provides control and configuration of system memory 140. The MCH 130 provides interface circuits to recognize and service isolated access assertions on memory reference bus cycles, including isolated memory read and write cycles. In addition, the MCH 130 has memory range registers (e.g., base and length registers) to represent the isolated area in the system memory 140. Once configured, the MCH 130

aborts any access to the isolated area that does not have the isolated access bus mode asserted.

The system memory 140 stores system code and data. The system memory 140 is typically implemented with dynamic random access memory (DRAM) or static random access memory (SRAM). The system memory 140 includes the accessible physical memory 60 (shown in Figure 1B). The accessible physical memory includes a loaded operating system 142, the isolated area 70 (shown in Figure 1B), and an isolated control and status space 148. The loaded operating system 142 is the portion of the operating system that is loaded into the system memory 140. The loaded OS 142 is typically loaded from a mass storage device via some boot code in a boot storage such as a boot read only memory (ROM). The isolated area 70, as shown in Figure 1B, is the memory area that is defined by the processor 110 when operating in the isolated execution mode. Access to the isolated area 70 is restricted and is enforced by the processor 110 and/or the MCH 130 or other chipset that integrates the isolated area functionalities. The isolated control and status space 148 is an input/output (I/O)-like, independent address space defined by the processor 110. The isolated control and status space 148 contains mainly the isolated execution control and status registers. The isolated control and status space 148 does not overlap any existing address space and is accessed using the isolated bus cycles. The system memory 140 may also include other programs or data that are not shown.

The ICH 150 represents a known single point in the system having the isolated execution functionality. For clarity, only one ICH 150 is shown. The system 100 may have many ICH's similar to the ICH 150. When there are multiple ICH's, a designated ICH is selected to control the isolated area configuration and status. In one embodiment, this selection is performed by an external strapping pin. As is known by one skilled in the art, other methods of selecting can be used, including using programmable configuring registers. The ICH 150 has a number of functionalities that are designed to support the isolated execution mode in addition to the traditional I/O functions. In particular, the ICH 150 includes an isolated bus cycle interface 152, the processor nub loader 52 (shown in Figure 1A), a digest memory 154, a cryptographic key storage 155, an isolated execution logical processor manager 156, and a token bus interface 159.

The isolated bus cycle interface 152 includes circuitry to interface to the isolated bus cycle signals to recognize and service isolated bus cycles, such as the isolated read

and write bus cycles. The processor nub loader 52, as shown in Figure 1A, includes a processor nub loader code and its digest (e.g., cryptographic hash) value. The processor nub loader 52 is invoked by execution of an appropriate isolated instruction (e.g., Iso_Init) and is transferred to the isolated area 70. From the isolated area 80, the processor nub loader 52 copies the processor nub 18 from the system flash memory (e.g., the processor nub code 18 in non-volatile memory 160) into the isolated area 70, verifies and logs its integrity, and manages a symmetric key used to protect the processor nub's secrets. In one embodiment, the processor nub loader 52 is implemented in read only memory (ROM). For security purposes, the processor nub loader 52 is unchanging, tamper-resistant and non-substitutable. The digest memory 154, typically implemented in RAM, stores the digest (e.g., cryptographic hash) values of the loaded processor nub 18, the operating system nub 16, and any other supervisory modules (e.g., ring-0 modules) loaded into the isolated execution space. The cryptographic key storage 155 holds a symmetric encryption/decryption key that is unique for the platform of the system 100. In one embodiment, the cryptographic key storage 155 includes internal fuses that are programmed at manufacturing. Alternatively, the cryptographic key storage 155 may also be created during manufacturing with a cryptographic random number generator. The isolated execution logical processor manager 156 manages the operation of logical processors configuring their isolated execution mode support. In one embodiment, the isolated execution logical processor manager 156 includes a logical processor count register that tracks the number of logical processors participating in the isolated execution mode. The token bus interface 159 interfaces to the token bus 180. A combination of the processor nub loader digest, the processor nub digest, the operating system nub digest, and optionally additional digests, represents the overall isolated execution digest, referred to as isolated digest. The isolated digest is a fingerprint identifying the all supervisory code involved in controlling the isolated execution configuration and operation. The isolated digest is used to attest or prove the state of the current isolated execution environment.

The non-volatile memory 160 stores non-volatile information. Typically, the non-volatile memory 160 is implemented in flash memory. In one embodiment, the non-volatile memory 160 includes the processor nub 18. The processor nub 18 provides set-up and low-level management of the isolated area 70 (in the system memory 140), including verification, loading, and logging of the operating system nub 16, and the management of the symmetric key used to protect the operating system

nub's secrets. The processor nub loader 52 performs some part of the setup and manages/updates the symmetric key before the processor nub 18 and the OS nub 16 are loaded. The processor nub 18 The processor nub 18 may also provide interface abstractions to low-level security services provided by other hardware. The processor nub 18 may also be distributed by the original equipment manufacturer (OEM) or operating system vendor (OSV).

The mass storage device 170 stores archive information such as code (e.g., processor nub 18), programs, files, data, applications (e.g., applications 42_i to 42_N), applets (e.g., applets 46_i to 46_K) and operating systems. The mass storage device 170 may include compact disk (CD) ROM 172, floppy diskettes 174, and hard drive 176, and any other storage devices. The mass storage device 170 provides a mechanism to read machine-readable media. When implemented in software, the elements of the present invention are the code segments to perform the necessary tasks. The program or code segments can be stored in a processor readable medium or transmitted by a computer data signal embodied in a carrier wave, or a signal modulated by a carrier, over a transmission medium. The "processor readable medium" may include any medium that can store or transfer information. Examples of the processor readable medium include an electronic circuit, a semiconductor memory device, a ROM, a flash memory, an erasable programmable ROM (EPROM), a floppy diskette, a compact disk CD-ROM, an optical disk, a hard disk, a fiber optical medium, a radio frequency (RF) link, etc. The computer data signal may include any signal that can propagate over a transmission medium such as electronic network channels, optical fibers, air, electromagnetic, RF links, etc. The code segments may be downloaded via computer networks such as the Internet, an Intranet, etc.

I/O devices 175 may include any I/O devices to perform I/O functions. Examples of I/O devices 175 include a controller for input devices (e.g., keyboard, mouse, trackball, pointing device), media card (e.g., audio, video, graphics), a network card, and any other peripheral controllers.

The token bus 180 provides an interface between the ICH 150 and various tokens in the system. A token is a device that performs dedicated input/output functions with security functionalities. A token has characteristics similar to a smart card, including at least one reserved-purpose public/private key pair and the ability to sign data with the private key. Examples of tokens connected to the token bus 180 include a motherboard token 182, a token reader 184, and other portable tokens 186

(e.g., smart card). The token bus interface 159 in the ICH 150 connects through the token bus 180 to the ICH 150 and ensures that when commanded to prove the state of the isolated execution, the corresponding token (e.g., the motherboard token 182, the token 186) signs only valid isolated digest information. For purposes of security, the token should be connected to the digest memory via the token bus 180.

A CHIPSET CIRCUIT TO MANAGE A SECURE PLATFORM

The overall architecture discussed above provides a basic insight into a hierarchical executive architecture to manage a secure platform. The elements shown in Figures 1A, 1B, and 1C are instances of an abstract model of this hierarchical executive architecture. The implementation of this hierarchical executive architecture is a combination of hardware and software. In what follows, the processor executive, the processor executive handler, and the operating system executive are abstract models of the processor nub 18, the processor nub loader 52, and the operating system nub 16 (Figures 1A, 1B, and 1C), respectively.

Figure 2 is a diagram illustrating a chipset environment 200 according to one embodiment of the invention. The chipset environment 200 includes a chipset 205 and executive entities 220.

The chipset 205 is an integrated device that provides the necessary infrastructure for the isolated execution mode in the platform. In one embodiment, the chipset 205 is the input/output controller hub (ICH) 150 shown in Figure 1C. The chipset 205 includes a chipset circuit 210. The chipset circuit 210 includes the functionalities to allow a platform to operate in a secure environment. The secure environment is associated with the isolated memory area (e.g., the isolated memory area 70 shown in Figure 1C) that is accessible to a processor in the platform. The platform may have one or multiple processors (e.g., the processor 110 in Figure 1C). Each processor may operate in one of a normal execution mode and an isolated execution mode. Each processor may have one or multiple logical threads. The chipset 205 keeps tracks of the number of logical threads operating in the execution mode for the entire platform, whether the platform is a uni- or multi-processor system.

The executive entities 220 include a number of executives that are designed to operate in the secure environment. The executive entities 220 are associated with the isolated memory area 70 (Figure 1C). The executive entities 220 include a processor executive (PE) handler 230, a PE 240, and an operating system executive (OSE) 250.

The PE handler 230 handles the PE 240. The PE 240 handles the OSE 250. The OSE 250 interfaces to a subset of the operating system (OS) 260 running on the platform. As mentioned above, the PE handler 230, the PE 240, and the OSE 250 are abstract models of the processor nub loader 52, the processor nub 18, and the operating system nub 16, respectively, shown in Figures 1A-1C.

Figure 3 is a diagram illustrating the chipset circuit 210 shown in Figure 2 according to one embodiment of the invention. The chipset circuit 210 includes an initialization storage 310, a PE handler storage 320, a thread count storage 330, a thread count updater 340, a mode storage 350, a mode write circuit 360, an identifier log storage 370, a log lock storage 375, a lock circuit 378, a platform key storage 380, a scratch storage 390, and a status storage 395. The initialization storage 310, the PE handler storage 320, the thread count storage 330, the mode storage 350, the identifier log storage 370, the log lock storage 375, the platform key storage 380, the scratch storage 390, and the status storage 390 are all mapped into an input/output address space accessible to the processor 110 in isolated execution mode. The thread count updater 340, the mode write circuit 360, and the lock circuit 378 are invoked or triggered when the corresponding storage is accessed.

The initialization storage 310 is used to initialize the chipset 205 for isolated execution mode. The initialization storage is accessed by an initialization storage access signal 305. The initialization storage access signal 305 represents an input/output bus access cycle decoded to the initialization storage address as generated by one of the processors in the system (e.g., processor 110 in Figure 1C).

The PE handler storage 320 stores PE handler data 322. The PE handler data 322 include a PE handler image 323, a PE handler cryptographic identifier 325, a PE handler size 327, and a PE handler address 329. The PE handler image 323 is to be loaded into the isolated memory area 70 after the chipset 205 is initialized. The loaded PE handler image 323 corresponds to the PE handler 230 in the executive entities 220 (Figure 2).

The thread count storage 330 stores a thread count 332 to indicate the number of threads currently initialized for operating in the isolated execution mode. The thread count 332 is useful for a thread to know if it is the first thread to initialize the platform in the isolated execution mode and whether it is the last thread to close the isolated area. The limit comparator 335 compares the thread count 332 with high and low limits to determine if the thread count has exceeded the maximum threads. When a maximum

limit is exceeded, a failure or fault condition is generated. When the thread count reaches a minimum limit (e.g., zero), the log lock storage 375 is initialized to its initial conditions. The updated thread count is one of an incremented thread count and a decremented thread count. The incremented thread count is returned when one of the
5 threads enrolls in the isolated execution mode. The decremented thread count is returned when one of the enrolled threads withdraws from the isolated execution mode.

The mode storage 350 stores a chipset mode 352. The mode circuit 360 writes the chipset mode 352 into the mode storage 350. The chipset mode 352 indicates a mode of operation of the chipset 205. This mode of operation includes an initialization
10 waiting mode, a PE initialization in-progress mode, a PE initialization completion mode, an OSE loaded mode, a closing mode, and a failure mode. The initialization waiting mode indicates that the chipset is waiting for initialization. The PE initialization in-progress mode indicates that the PE is being executed. The PE initialization completion mode indicates that the PE is completed. The OSE loaded
15 mode indicates the OSE has been loaded. The closing mode indicates that the isolated execution mode is closed. The failure mode indicates that there is a failure.

The identifier log storage 370 stores cryptographic identifiers 372_1 to 372_N of the executive entities loaded in the isolated execution mode even if they are later loaded. The cryptographic identifiers 372_1 to 372_N are read-only when in lock. The
20 log lock storage 375 stores a lock pattern 376 to indicate which of the cryptographic identifiers 372_1 to 372_N is in lock. To be "in lock" means that the corresponding identifier cannot be modified or written. The lock pattern 376 is initialized when the thread count reaches the minimum limit. The cryptographic identifiers 372_1 to 372_N are locked based on the lock pattern 376.

The platform key storage 380 stores a platform key 385. The platform key 385 is used in handling the executive entities 220. The platform key 385 is returned when the platform key storage 380 is read in an initialization waiting mode as set by the mode
25 storage 350 which will be described later. The platform key 385 is programmed at manufacturing time to a random value 387.

The scratch storage 390 stores isolated settings 392 used to configure the isolated execution mode. The isolated settings 392 include an isolated base value, an isolated length value, and a processor executive entry address. The isolated base and length values define the isolated memory area 70, and are essentially the same as the
30

isolated base and length values as stored in the processor 110 and other chipsets (e.g., the MCH 130 in Figure 1C).

The status storage 395 stores a status value 396 of an isolated unlock pin 397.

Figure 4 is a flowchart illustrating a process 400 to manage a chipset according to one embodiment of the invention.

Upon START, the process 400 boots up the platform and configures the platform in the isolated execution mode (Block 410). Then, the process 400 writes the chipset mode as the initialization waiting mode (Block 420). Then, the process 400 determines if the thread operation is a thread enrolment or a thread withdrawal (Block 430). If it is a thread enrolment, the process 400 enrolls the thread (Block 440). If it is a thread withdrawal, the process 400 withdraws the thread (Block 450) and is then terminated.

After the process 400 enrolls the thread, the process 400 loads the PE handler image into the isolated memory area (Block 460). The loaded PE handler image becomes the PE handler. Next, the process 400 initializes, invokes PE and OSE and write chipset mode (Block 470). Then, the process 400 is terminated.

Figure 5 is a flowchart illustrating the process 440 to enroll a thread shown in Figure 4 according to one embodiment of the invention.

Upon START, the process 440 accesses the enrollment storage (Block 510). The process 440 determines if the thread count has reached the high limit or if the error mode is set (Block 520). If yes, the process 440 writes the chipset mode as a failure mode (Block 530), returns the current thread count (Block 540) and is terminated. Otherwise, the process 440 increments the thread count (Block 550). The process 440 then determines if failure mode is set (Block 560). If yes, the process 440 returns the current thread count (Block 540). Otherwise, the process 440 returns incremented thread count (Block 570). The process 440 is terminated.

Figure 6 is a flowchart illustrating the process 450 to withdraw a thread shown in Figure 4 according to one embodiment of the invention.

Upon START, the process 450 accesses the withdrawal storage (Block 610). The process 450 checks to see if the error/failure mode is set (Block 615). If yes, the process 450 returns the decrement thread count current (Block 660). The process 450 is terminated. Otherwise, the process 450 decrements the thread count (Block 620). Then, the process 450 determines if the thread count has reached a minimum limit (Block 630). If no, the process 450 returns the decremented thread count (Block 660)

CLAIMS

What is claimed is:

1. An apparatus comprising:

an initialization storage to initialize a chipset in a secure environment for an isolated execution mode, the secure environment having a plurality of executive entities and being associated with an isolated memory area accessible by at least one processor, the at least one processor having a plurality of threads and operating in one of a normal execution mode and the isolated execution mode, the executive entities including a processor executive (PE) handler; and

a PE handler storage to store PE handler data corresponding to the PE handler, the PE handler data including a PE handler image to be loaded into the isolated memory area after the chipset is initialized, the loaded PE handler image corresponding to the PE handler.

2. The apparatus of claim 1 further comprises:

a thread count storage to store a thread count indicating number of threads currently initialized for operation in the isolated execution mode;

a thread count updater coupled to the thread count storage to update the thread count;

a mode storage to store a chipset mode indicating a mode of operation of the chipset; and

a mode write circuit coupled to the mode storage to write the chipset mode to the mode storage.

3. The apparatus of claim 2 further comprising:

an identifier log storage to store cryptographic identifiers of the executive entities loaded into the isolated execution mode, the cryptographic identifiers being read only when in lock;

a log lock storage to store a lock pattern indicating the identifiers in lock; and

a lock circuit coupled to the identifier log storage and the log lock storage to lock the identifiers based on the lock pattern.

1 4. The apparatus of claim 3 further comprising:
2 a platform key storage to store a platform key used in handling the executive
3 entities; and
4 a scratch storage to store isolated settings used to configure the isolated
5 execution mode.

1 5. The apparatus of claim 4 wherein the executive entities further include a
2 processor executive (PE) and an operating system executive (OSE).

1 6. The apparatus of claim 5 wherein the chipset mode is one of an
2 initialization waiting mode to indicate the chipset is waiting for initialization, a PE
3 initialization in-progress mode to indicate the PE is being executed, a PE initialization
4 completion mode to indicate the PE is completed, an OSE loaded mode to indicate the
5 OSE has been loaded, a closing mode to indicate the isolated execution mode is closed,
6 and a failure mode to indicate a failure.

1 7. The apparatus of claim 6 wherein the initialization storage returns an
2 updated thread count when the chipset mode does not represent the failure mode and to
3 return a current thread count when the chipset mode represents the failure mode, the
4 updated thread count being one of an incremented thread count and a decremented
5 thread count.

1 8. The apparatus of claim 7 wherein the initialization storage comprises:
2 an enrollment storage to return the incremented thread count when one of the
3 threads enrolls in the isolated execution mode; and
4 a withdrawal storage to return the decremented thread count when one of the
5 enrolled threads withdraws from the isolated execution mode.

1 9. The apparatus of claim 8 wherein the mode write circuit writes the
2 chipset mode corresponding to a failure mode into the mode storage when the thread
3 count reaches a thread limit.

1 10. The apparatus of claim 1 wherein the PE handler data further include a
2 cryptographic PE handler identifier, a PE handler size, and a PE handler address.

1 11. The apparatus of claim 6 wherein the PE handler storage is a read-only
2 memory.

1 12. The apparatus of claim 6 wherein the platform key is returned when the
2 platform key storage is read in the initialization waiting mode.

1 13. The apparatus of claim 12 wherein the platform key is programmed to a
2 random value.

1 14. The apparatus of claim 13 further comprising:
2 a status storage to store a status value of an isolated unlock pin used in setting
3 platform settings.

1 15. The apparatus of claim 4 wherein the isolated settings include an
2 isolated base value, an isolated length value, and a processor executive entry address,
3 the isolated base and length values defining the isolated memory area.

1 16. A method comprising:
2 initializing a chipset in a secure environment for an isolated execution mode by
3 an initialization storage, the secure environment having a plurality of executive entities
4 and being associated with an isolated memory area accessible by at least one processor,
5 the at least one processor having a plurality of threads and operating in one of a normal
6 execution mode and the isolated execution mode, the executive entities including a
7 processor executive (PE) handler; and
8 storing PE handler data corresponding to the PE handler in a PE handler storage,
9 the PE handler data including a PE handler image to be loaded into the isolated memory
10 area after the chipset is initialized, the loaded PE handler image corresponding to the
11 PE handler.

1 17. The method of claim 16 further comprises:
2 storing a thread count in a thread count storage indicating number of threads
3 currently initialized for operation in the isolated execution mode;
4 updating the thread count when the initialization storage is accessed;
5 storing a chipset mode indicating a mode of operation of the chipset in a mode
6 storage; and
7 writing the chipset mode into the mode storage.

1 18. The method of claim 17 further comprising:
2 storing cryptographic identifiers of the executive entities loaded into the isolated
3 execution mode, the identifiers being read only when in lock;
4 storing a lock pattern indicating the identifiers in lock; and
5 locking the identifiers based on the lock pattern.

1 19. The method of claim 18 further comprising:
2 storing a platform key used in handling the executive entities in a platform key
3 storage; and
4 storing isolated settings used to configure the isolated execution mode.

1 20. The method of claim 19 wherein the executive entities further include a
2 processor executive (PE) and an operating system executive (OSE).

1 21. The method of claim 20 wherein the chipset mode is one of an
2 initialization waiting mode to indicate the chipset is waiting for initialization, a PE
3 initialization in-progress mode to indicate the PE is being executed, a PE initialization
4 completion mode to indicate the PE is completed, an OSE loaded mode to indicate the
5 OSE has been loaded, a closing mode to indicate the isolated execution mode is closed,
6 and a failure mode to indicate a failure.

1 22. The method of claim 21 wherein initializing the chipset comprises

2 returning an updated thread count when the chipset mode does not represent the
3 failure mode, the updated thread count being one of an incremented thread count and a
4 decremented thread count; and

5 returning a current thread count when the chipset mode represents the failure
6 mode.

1 23. The method of claim 22 wherein initializing the chipset further
2 comprises:

3 returning the incremented thread count when one of the threads enrolls in the
4 isolated execution mode; and

5 returning the decremented thread count when one of the enrolled threads
6 withdraws from the isolated execution mode.

1 24. The method of claim 23 wherein writing the chipset mode comprises
2 writing the chipset mode corresponding to a failure mode when the thread count reaches
3 a thread limit.

1 25. The method of claim 16 wherein the PE handler data further include a
2 PE handler identifier, a PE handler size, and a PE handler address.

1 26. The method of claim 21 wherein the PE handler storage is read-only
2 memory.

1 27. The method of claim 21 wherein the platform key is returned when the
2 platform key storage is read in the initialization waiting mode.

1 28. The method of claim 27 wherein the platform key is programmed to a
2 random value.

1 29. The method of claim 28 further comprising:
2 storing a status value of an isolated unlock pin used to unlock and allow
3 platform setting.

30. The method of claim 19 wherein the isolated settings include an isolated base value, an isolated length value, and a processor executive entry address, the isolated base and length values defining the isolated memory area.

31. A computer program product comprising:
a machine useable medium having computer program code embedded therein, the computer program product having:
computer readable program code for initializing a chipset in a secure environment for an isolated execution mode by an initialization storage, the secure environment having a plurality of executive entities and being associated with an isolated memory area accessible by at least one processor, the at least one processor having a plurality of threads and operating in one of a normal execution mode and the isolated execution mode, the executive entities including a processor executive (PE) handler; and
computer readable program code for storing PE handler data corresponding to the PE handler in a PE handler storage, the PE handler data including a PE handler image to be loaded into the isolated memory area after the chipset is initialized, the loaded PE handler image corresponding to the PE handler.

32. The computer program product of claim 31 further comprises:
computer readable program code for storing a thread count in a thread count storage indicating number of threads currently initialized for operation in the isolated execution mode;
computer readable program code for updating the thread count when the initialization storage is accessed;
computer readable program code for storing a chipset mode indicating a mode of operation of the chipset in a mode storage; and
computer readable program code for writing the chipset mode into the mode storage.

33. The computer program product of claim 32 further comprising:

2 computer readable program code for storing cryptographic identifiers of the
3 executive entities loaded into the isolated execution mode, the identifiers being read
4 only when in lock;

5 computer readable program code for storing a lock pattern indicating the
6 identifiers in lock; and

7 computer readable program code for locking the identifiers based on the lock
8 pattern.

1 34. The computer program product of claim 33 further comprising:
2 computer readable program code for storing a platform key used in handling the
3 executive entities in a platform key storage; and
4 computer readable program code for storing isolated settings used to configure
5 the isolated execution mode.

1 35. The computer program product of claim 34 wherein the executive
2 entities further include a processor executive (PE) and an operating system executive
3 (OSE).

1 36. The computer program product of claim 35 wherein the chipset mode is
2 one of an initialization waiting mode to indicate the chipset is waiting for initialization,
3 a PE initialization in-progress mode to indicate the PE is being executed, a PE
4 initialization completion mode to indicate the PE is completed, an OSE loaded mode to
5 indicate the OSE has been loaded, a closing mode to indicate the isolated execution
6 mode is closed, and a failure mode to indicate a failure.

1 37. The computer program product of claim 36 wherein the computer
2 readable program code for initializing the chipset comprises
3 computer readable program code for returning an updated thread count when the
4 chipset mode does not represent the failure mode, the updated thread count being one of
5 an incremented thread count and a decremented thread count; and
6 computer readable program code for returning a current thread count when the
7 chipset mode represents the failure mode.

1 38. The computer program product of claim 37 wherein the computer
2 readable program code for initializing the chipset further comprises:
3 computer readable program code for returning the incremented thread count
4 when one of the threads enrolls in the isolated execution mode; and
5 computer readable program code for returning the decremented thread count
6 when one of the enrolled threads withdraws from the isolated execution mode.

1 39. The computer program product of claim 38 wherein the computer
2 readable program code for writing the chipset mode comprises computer readable
3 program code for writing the chipset mode corresponding to a failure mode when the
4 thread count reaches a thread limit.

1 40. The computer program product of claim 31 wherein the PE handler data
2 further include a PE handler cryptographic identifier, a PE handler size, and a PE
3 handler address.

1 41. The computer program product of claim 36 wherein the PE handler
2 storage is a read-only memory.

1 42. The computer program product of claim 36 wherein the platform key is
2 returned when the platform key storage is read in the initialization waiting mode.

1 43. The computer program product of claim 42 wherein the platform key is
2 programmed to a random value.

1 44. The computer program product of claim 43 further comprising:
2 computer readable program code for storing a status value of an isolated unlock
3 pin used to unlock and allow platform settings.

1 45. The computer program product of claim 34 wherein the isolated settings
2 include an isolated base value, an isolated length value, and a processor executive entry
3 address, the isolated base and length values defining the isolated memory area.

1 46. A system comprising:
2 at least one processor having a plurality of threads and operating in one of a
3 normal execution mode and an isolated execution mode;
4 a memory having an isolated memory area accessible to the at least one
5 processor in the isolated execution mode; and
6 a chipset circuit coupled to the at least one processor and the memory
7 comprising:
8 an initialization storage to initialize a chipset in a secure environment for the
9 isolated execution mode, the secure environment having a plurality of executive entities
10 and being associated with the isolated memory area, the executive entities including a
11 processor executive (PE) handler, and
12 a PE handler storage to store PE handler data corresponding to the PE handler,
13 the PE handler data including a PE handler image to be loaded into the isolated memory
14 area after the chipset is initialized, the loaded PE handler image corresponding to the
15 PE handler.

1 47. The system of claim 46 wherein the chipset circuit further comprises:
2 a thread count storage to store a thread count indicating number of threads
3 currently operating in the isolated execution mode,
4 a thread count updater coupled to the thread count storage to update the thread
5 count when the initialization storage is accessed;
6 a mode storage to store a chipset mode indicating a mode of operation of the
7 chipset; and
8 a mode write circuit coupled to the mode storage to write the chipset mode into
9 the mode storage.

1 48. The system of claim 47 wherein the chipset circuit further comprises:

2 an identifier log storage to store cryptographic identifiers of the executive
3 entities operating in the isolated execution mode, the identifiers being read only when
4 in lock;

5 a log lock storage to store a lock pattern indicating the identifiers in lock; and

6 a lock circuit coupled to the identifier log storage and the log lock storage to
7 lock the identifiers based on the lock pattern.

1 49. The system of claim 48 wherein the chipset circuit further comprises:

2 a platform key storage to store a platform key used in handling the executive
3 entities; and

4 a scratch storage to store isolated settings used to configure the isolated
5 execution mode.

1 50. The system of claim 49 wherein the executive entities further include a
2 processor executive (PE) and an operating system executive (OSE).

1 51. The system of claim 50 wherein the chipset mode is one of an
2 initialization waiting mode to indicate the chipset is waiting for initialization, a PE
3 initialization in-progress mode to indicate the PE is being executed, a PE initialization
4 completion mode to indicate the PE is completed, an OSE loaded mode to indicate the
5 OSE has been loaded, a closing mode to indicate the isolated execution mode is closed,
6 and a failure mode to indicate a failure.

1 52. The system of claim 51 wherein the initialization storage returns an
2 updated thread count when the chipset mode does not represent the failure mode and to
3 return a current thread count when the chipset mode represents the failure mode, the
4 updated thread count being one of an incremented thread count and a decremented
5 thread count.

1 53. The system of claim 52 wherein the initialization storage comprises:

2 an enrollment storage to return the incremented thread count when one of the
3 threads enrolls in the isolated execution mode; and

4 a withdrawal storage to return the decremented thread count when one of the
5 enrolled threads withdraws from the isolated execution mode.

1 54. The system of claim 53 wherein the mode write circuit writes the chipset
2 mode corresponding to a failure mode into the mode storage when the thread count
3 reaches a thread limit.

1 55. The system of claim 46 wherein the PE handler data further include a PE
2 handler cryptographic identifier, a PE handler size, and a PE handler address.

1 56. The system of claim 51 wherein the PE handler storage is a non-volatile
2 memory.

1 57. The system of claim 51 wherein the platform key is returned when the
2 platform key storage is read in the initialization waiting mode.

1 58. The system of claim 57 wherein the platform key is programmed to a
2 random value.

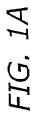
1 59. The system of claim 58 wherein the chipset circuit further comprises:
2 a status storage to store a status value of an isolated unlock pin used to unlock
3 and allow platform settings.

1 60. The system of claim 49 wherein the isolated settings include an isolated
2 base value, an isolated length value, and a processor executive entry address, the
3 isolated base and length values defining the isolated memory area.

ABSTRACT OF THE DISCLOSURE

A chipset is initialized in a secure environment for an isolated execution mode by an initialization storage. The secure environment has a plurality of executive entities and is associated with an isolated memory area accessible by at least one processor.

- 5 The at least one processor has a plurality of threads and operates in one of a normal execution mode and the isolated execution mode. The executive entities include a processor executive (PE) handler. PE handler data corresponding to the PE handler are stored in a PE handler storage. The PE handler data include a PE handler image to be loaded into the isolated memory area after the chipset is initialized. The loaded PE
- 10 handler image corresponds to the PE handler.



100

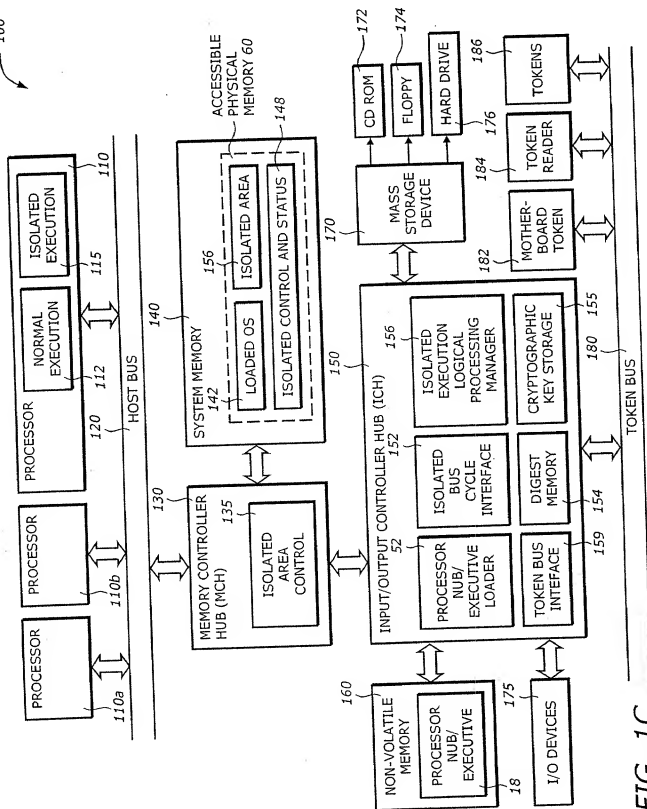


FIG. 1C

200

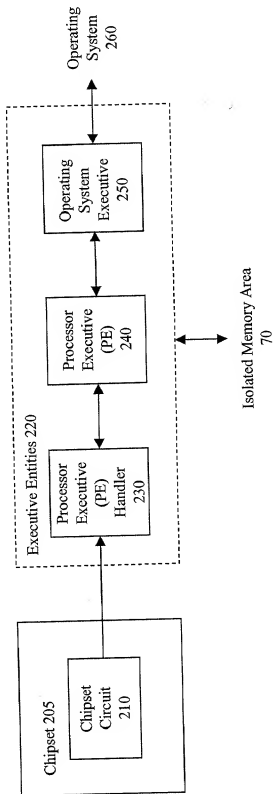
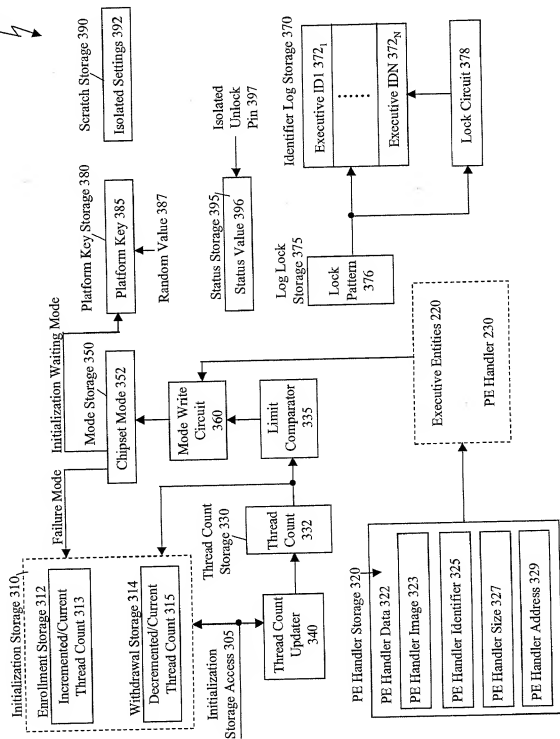


Fig. 2



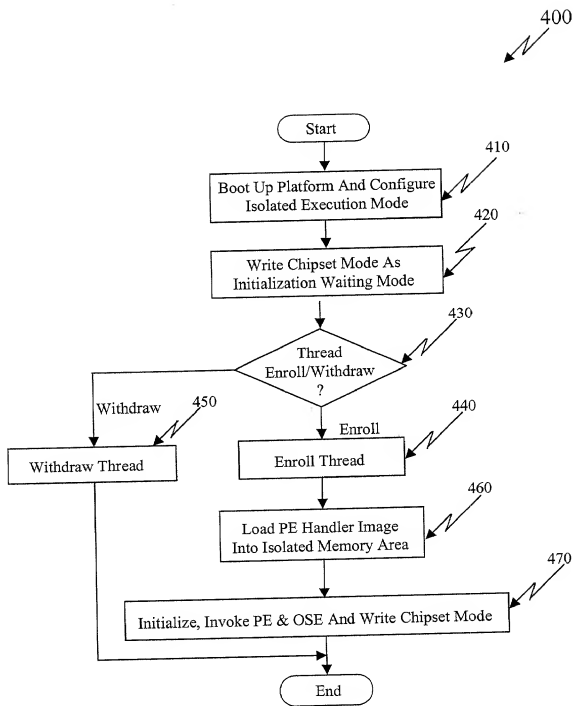


Fig. 4

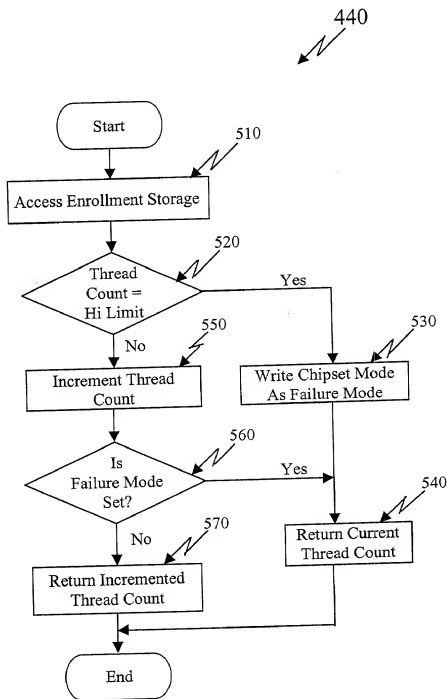


Fig. 5

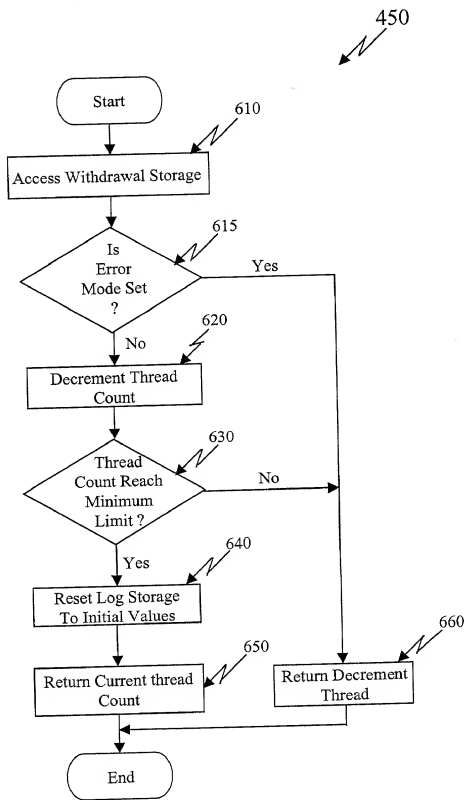


Fig. 6

**DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION
(CONTINUATION-IN-PART)**

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below, next to my name.

I believe I am the original, first, and sole inventor (if only one name is listed below) or any original, first, and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

MANAGING A SECURE ENVIRONMENT USING A CHIPSET IN ISOLATED EXECUTION MODE

the specification of which

☒ is attached hereto.
☐ was filed on _____ as
United States Application Number _____
or PCT International Application Number _____
and was amended on _____
(if applicable)

That this application in part discloses and claims subject matter disclosed in my earlier filed pending application:

Application No.: 09/540,613
Filed: 3/31/2000

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

That as to the subject matter of this application which is common to said earlier application, I do not know and do not believe that the same was ever known or used in the United States of America before my invention thereof, or patented or described in any printed publication in any country before my invention thereof or more than one year prior to this application, that the same was not in public use or on sale in the United States of America more than one year prior to this application, and

That said common subject matter has not been patented or made the subject of an inventor's certificate issued before the date of said earlier application in any country foreign to the United States of America on an application filed by me or my legal representatives or assigns more than twelve months prior to said earlier application;

I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, Section 119, of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

APPLICATION NUMBER	COUNTRY (OR INDICATE IF PCT)	DATE OF FILING (day, month, year)	PRIORITY CLAIMED UNDER 37 USC 119
			<input type="checkbox"/> No <input type="checkbox"/> Yes
			<input type="checkbox"/> No <input type="checkbox"/> Yes
			<input type="checkbox"/> No <input type="checkbox"/> Yes

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, Section 112, I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application:

APPLICATION NUMBER	FILING DATE	STATUS (ISSUED, PENDING, ABANDONED)

I hereby appoint the persons listed on Appendix A hereto (which is incorporated by reference and a part of this document) as my respective patent attorneys and patent agents, with full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office connected herewith.

Send correspondence to:

Thinh V. Nguyen, Reg. No. 42,034, BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN, LLP
(Name of Attorney or Agent)
12400 Wilshire Boulevard, 7th Floor, Los Angeles, California 90025 and direct telephone calls to:
Thinh V. Nguyen, (714) 557-3800.
(Name of Attorney or Agent)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of Sole/First Inventor (given name, family name) Carl M. Ellison

Inventor's Signature Carl M. Ellison Date 8/2/00

Residence Portland, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 1818 N.W. 28th Avenue
Portland, Oregon 97210-2214 USA

Full Name of Second/Joint Inventor (given name, family name) Roger A. Golliver

Inventor's Signature _____ Date _____

Residence Beaverton, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 16185 SE Night Hawk Drive
Beaverton, Oregon 97007-8368 USA

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, Section 112, I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application:

APPLICATION NUMBER	FILING DATE	STATUS (ISSUED, PENDING, ABANDONED)

I hereby appoint the persons listed on Appendix A hereto (which is incorporated by reference and a part of this document) as my respective patent attorneys and patent agents, with full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office connected herewith.

Send correspondence to:

Thinh V. Nguyen, Reg. No. 42,034, BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN, LLP
(Name of Attorney or Agent)
12400 Wilshire Boulevard, 7th Floor, Los Angeles, California 90025 and direct telephone calls to:
Thinh V. Nguyen, (714) 557-3800.
(Name of Attorney or Agent)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of Sole/First Inventor (given name, family name) Carl M. Ellison

Inventor's Signature _____ Date _____

Residence Portland, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 1818 N.W. 28th Avenue
Portland, Oregon 97210-2214 USA

Full Name of Second/Joint Inventor (given name, family name) Roger A. Golliver

Inventor's Signature Roger A. Golliver Date 7/28/2008

Residence Beaverton, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 16185 SE Night Hawk Drive
Beaverton, Oregon 97007-8368 USA

Full Name of Third/Joint Inventor (given name, family name) Howard C. Herbert
Inventor's Signature Howard C. Herbert **Date** 24 July 2000
Residence Phoenix, Arizona USA **Citizenship** USA
(City, State) (Country)
P. O. Address 16817 South 1st Drive
Phoenix, Arizona 85045 USA

Full Name of Fourth/Joint Inventor (given name, family name) Derrick C. Lin
Inventor's Signature _____ **Date** _____
Residence Foster City, California USA **Citizenship** USA
(City, State) (Country)
P. O. Address 113 Barkentine Street
Foster City, California 94404 USA

Full Name of Fifth/Joint Inventor (given name, family name) Francis X. McKeen
Inventor's Signature _____ **Date** _____
Residence Portland, Oregon USA **Citizenship** USA
(City, State) (Country)
P. O. Address 10612 N. W. LeMans Ct.
Portland, Oregon 97229 USA

Full Name of Sixth/Joint Inventor (given name, family name) Gilbert N. Neiger
Inventor's Signature _____ **Date** _____
Residence Portland, Oregon USA **Citizenship** USA
(City, State) (Country)
P. O. Address 2424 N. E. 11th Avenue
Portland, Oregon 97212 USA

Full Name of Third/Joint Inventor (given name, family name) Howard C. Herbert

Inventor's Signature _____ Date _____

Residence Phoenix, Arizona USA Citizenship USA
(City, State) (Country)

P. O. Address 16817 South 1st Drive
Phoenix, Arizona 85045 USA

Full Name of Fourth/Joint Inventor (given name, family name) Derrick C. Lin

Inventor's Signature  Date 8/30/00

Residence San Mateo, California USA Citizenship USA
(City, State) (Country)

P. O. Address 1737 Oakwood Drive
San Mateo, California 9440 USA

Full Name of Fifth/Joint Inventor (given name, family name) Francis X. McKeen

Inventor's Signature _____ Date _____

Residence Portland, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 10612 N. W. LeMans Ct.
Portland, Oregon 97229 USA

Full Name of Sixth/Joint Inventor (given name, family name) Gilbert Neiger

Inventor's Signature _____ Date _____

Residence Portland, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 2424 N. E. 11th Avenue
Portland, Oregon 97212 USA

Full Name of Third/Joint Inventor (given name, family name) Howard C. Herbert

Inventor's Signature _____ Date _____

Residence Phoenix, Arizona USA Citizenship USA
(City, State) (Country)

P. O. Address 16817 South 1st Drive
Phoenix, Arizona 85045 USA

Full Name of Fourth/Joint Inventor (given name, family name) Derrick C. Lin

Inventor's Signature _____ Date _____

Residence Foster City, California USA Citizenship USA
(City, State) (Country)

P. O. Address 113 Barkentine Street
Foster City, California 94404 USA

Full Name of Fifth/Joint Inventor (given name, family name) Francis X. McKeen

Inventor's Signature Francis X. McKeen Date 8/3/00

Residence Portland, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 10612 N. W. LeMans Ct.
Portland, Oregon 97229 USA

Full Name of Sixth/Joint Inventor (given name, family name) Gilbert N. Neiger

Inventor's Signature _____ Date _____

Residence Portland, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 2424 N. E. 11th Avenue
Portland, Oregon 97212 USA

Full Name of Third/Joint Inventor (given name, family name) Howard C. Herbert

Inventor's Signature _____ Date _____

Residence Phoenix, Arizona USA Citizenship USA
(City, State) (Country)

P. O. Address 16817 South 1st Drive
Phoenix, Arizona 85045 USA

Full Name of Fourth/Joint Inventor (given name, family name) Derrick C. Lin

Inventor's Signature _____ Date _____

Residence Foster City, California USA Citizenship USA
(City, State) (Country)

P. O. Address 113 Barkentine Street
Foster City, California 94404 USA

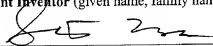
Full Name of Fifth/Joint Inventor (given name, family name) Francis X. McKeen

Inventor's Signature _____ Date _____

Residence Portland, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 10612 N. W. LeMans Ct.
Portland, Oregon 97229 USA

Full Name of Sixth/Joint Inventor (given name, family name) Gilbert Neiger

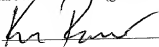
Inventor's Signature  Date 7/27/2000

Residence Portland, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 2424 N. E. 11th Avenue
Portland, Oregon 97212 USA

Full Name of Seventh/Joint Inventor (given name, family name)

Ken Reneris

Inventor's Signature 

Date

8/3/2000

Residence Wilbraham, Massachusetts USA

(City, State)

Citizenship USA

(Country)

P. O. Address 8 Red Gap Road

Wilbraham, Massachusetts 01095 USA

Full Name of Eighth/Joint Inventor (given name, family name)

James A. Sutton

Inventor's Signature _____

Date _____

Residence Portland, Oregon USA

(City, State)

Citizenship USA

(Country)

P. O. Address 20205 N. W. Paulina Drive

Portland, Oregon 97229 USA

Full Name of Ninth/Joint Inventor (given name, family name)

Shreekant S. Thakkar

Inventor's Signature _____

Date _____

Residence Portland, Oregon USA

(City, State)

Citizenship United Kingdom

(Country)

P. O. Address 150 S.W. Moonridge Place

Portland, Oregon 92775 USA

Full Name of Tenth/Joint Inventor (given name, family name)

Millind Mittal

Inventor's Signature _____

Date _____

Residence Palo Alto, CA USA

(City, State)

Citizenship USA

(Country)

P. O. Address 800 E. Charleston Road, #29

Palo Alto, CA 94303 USA

Full Name of Seventh/Joint Inventor (given name, family name) Ken Reneris

Inventor's Signature _____ Date _____

Residence Wilbraham, Massachusetts USA Citizenship USA
(City, State) (Country)

P. O. Address 8 Red Gap Road
Wilbraham, Massachusetts 01095 USA

Full Name of Eighth/Joint Inventor (given name, family name) James A. Sutton

Inventor's Signature James A. Sutton Date 7-24-2000

Residence Portland, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 20205 N. W. Paulina Drive
Portland, Oregon 97229 USA

Full Name of Ninth/Joint Inventor (given name, family name) Shreekant S. Thakkar

Inventor's Signature _____ Date _____

Residence Portland, Oregon USA Citizenship United Kingdom
(City, State) (Country)

P. O. Address 150 S.W. Moonridge Place
Portland, Oregon 92775 USA

Full Name of Tenth/Joint Inventor (given name, family name) Millind Mittal

Inventor's Signature _____ Date _____

Residence Palo Alto, CA USA Citizenship USA
(City, State) (Country)

P. O. Address 800 E. Charleston Road, #29
Palo Alto, CA 94303 USA

Full Name of Seventh/Joint Inventor (given name, family name) Ken Reneris

Inventor's Signature _____ Date _____

Residence Wilbraham, Massachusetts USA Citizenship USA
(City, State) (Country)

P. O. Address 8 Red Gap Road
Wilbraham, Massachusetts 01095 USA

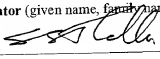
Full Name of Eighth/Joint Inventor (given name, family name) James A. Sutton

Inventor's Signature _____ Date _____

Residence Portland, Oregon USA Citizenship USA
(City, State) (Country)

P. O. Address 20205 N. W. Paulina Drive
Portland, Oregon 97229 USA

Full Name of Ninth/Joint Inventor (given name, family name) Shreekant S. Thakkar

Inventor's Signature  Date 8/7/00

Residence Portland, Oregon USA Citizenship United Kingdom
(City, State) (Country)

P. O. Address 150 S.W. Moonridge Place
Portland, Oregon 92775 USA

Full Name of Tenth/Joint Inventor (given name, family name) Millind Mittal

Inventor's Signature _____ Date _____

Residence Palo Alto, CA USA Citizenship USA
(City, State) (Country)

P. O. Address 800 E. Charleston Road, #29
Palo Alto, CA 94303 USA

PT: 2000

000E LSS 712

SS: 11 0002 50 dBS

Full Name of Tenth/Joint Inventor (given name, family name)

Millind Mittal

Inventor's Signature

Date

9/5/00

Residence Palo Alto, CA USA

Citizenship USA

(City, State)

(Country)

P. O. Address 800 E. Charleston Road, #29

Palo Alto, CA 94303 USA

Full Name of Eleventh/Joint Inventor (given name, family name)

Inventor's Signature

Date

Residence

Citizenship

(City, State)

(Country)

P. O. Address

Appendix A

I hereby appoint BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP, a firm including: William E. Alford, Reg. No. 37,764; Farzad E. Amini, Reg. No. 42,261; Aloysius T. C. AuYeung, Reg. No. 35,432; William Thomas Babbitt, Reg. No. 39,591; Carol F. Barry, Reg. No. 41,600; Jordan Michael Becker, Reg. No. 39,602; Lisa N. Benado, Reg. No. 39,995; Bradley J. Berezna, Reg. No. 33,474; Michael A. Bernadieu, Reg. No. 35,934; Roger W. Blakely, Jr., Reg. No. 25,831; R. Alan Burnett, Reg. No. 46,149; Gregory D. Caldwell, Reg. No. 39,926; Andrew C. Chen, Reg. No. 43,544; Thomas M. Coester, Reg. No. 39,637; Donna Jo Coningsby, Reg. No. 41,684; Florin Corie, Reg. No. 46,244; Dennis M. deGuzman, Reg. No. 41,702; Stephen M. De Klerk, Reg. No. P46,503; Michael Anthony DeSanctis, Reg. No. 39,957; Daniel M. De Vos, Reg. No. 37,813; Robert Andrew Diehl, Reg. No. 40,992; Sanjeet Dutta, Reg. No. P46,145; Matthew C. Fagan, Reg. No. 37,542; Tarek N. Fahmi, Reg. No. 41,402; George Fountain, Reg. No. 37,374; Paramita Ghosh, Reg. No. 42,806; James Y. Go, Reg. No. 40,621; James A. Henry, Reg. No. 41,064; Willmore F. Holbrow III, Reg. No. P41,845; Sheryl Sue Holloway, Reg. No. 37,850; George W. Hoover II, Reg. No. 32,992; Eric S. Hyman, Reg. No. 30,139; William W. Kidd, Reg. No. 31,772; Sang Hui Kim, Reg. No. 40,450; Walter T. Kim, Reg. No. 42,731; Eric T. King, Reg. No. 44,188; Erica W. Kuo, Reg. No. 42,775; George Brian Leavell, Reg. No. 45,436; Gordon R. Lindeen III, Reg. No. 33,192; Jan Carol Little, Reg. No. 41,181; Kurt P. Leyendecker, Reg. No. 42,799; Joseph Lutz, Reg. No. 43,765; Michael J. Mallie, Reg. No. 36,591; Andre L. Marais, under 37 C.F.R. § 10.9(b); Paul A. Mendonsa, Reg. No. 42,879; Clive D. Menezes, Reg. No. 45,493; Chun M. Ng, Reg. No. 36,878; Thien T. Nguyen, Reg. No. 43,835; Thinh V. Nguyen, Reg. No. 42,034; Dennis A. Nicholls, Reg. No. 42,036; Daniel E. Ovanezan, Reg. No. 41,236; Kenneth B. Paley, Reg. No. 38,989; Marina Portnova, Reg. No. P45,750; William F. Ryann, Reg. No. 44,313; James H. Salter, Reg. No. 35,668; William W. Schaal, Reg. No. 39,018; James C. Scheller, Reg. No. 31,195; Jeffrey Sam Smith, Reg. No. 39,377; Maria McCormack Sobrino, Reg. No. 31,639; Stanley W. Sokoloff, Reg. No. 25,128; Judith A. Szepesi, Reg. No. 39,393; Vincent P. Tassinari, Reg. No. 42,179; Edwin H. Taylor, Reg. No. 25,129; John F. Travis, Reg. No. 43,203; Joseph A. Twarowski, Reg. No. 42,191; Tom Van Zandt, Reg. No. 43,219; Lester J. Vincent, Reg. No. 31,460; Glenn E. Von Tersch, Reg. No. 41,364; John Patrick Ward, Reg. No. 40,216; Mark L. Watson, Reg. No. P46,322; Thomas C. Webster, Reg. No. P46,154; Steven D. Yates, Reg. No. 42,242; and Norman Zafman, Reg. No. 26,250; my patent attorneys, and Firasat Ali, Reg. No. 45,715; and Justin M. Dillon, Reg. No. 42,486; my patent agents, with offices located at 12400 Wilshire Boulevard, 7th Floor, Los Angeles, California 90025, telephone (714) 557-3800, with full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office connected herewith.